

AMENDMENTS TO THE CLAIMS

Claims 1-6, 8-13, 15-20, and 22-29 are pending. Please amend claims 1 and 6. No claims are canceled, added, or withdrawn.

The following listing of claims replaces all prior versions, and listings of claims in the application.

1. (Currently amended) A computer-implemented method for providing thread scheduling in a device, the device comprising one or more hardware elements operatively coupled to an operating system comprising a plurality of program modules, the method comprising:

scheduling one or more threads according to a predetermined periodic rate;

determining whether or not there are any threads to execute; and

responsive to a determination that there are no threads to execute, deactivating one or more of the hardware elements and the program modules for a dynamic variable amount of time, the dynamic variable amount of time being independent of the predetermined periodic rate and being based on a sleep state of a set of threads in a sleep queue.

2. (Original) A method as recited in claim 1, wherein the dynamic variable amount of time is based on a maximum amount of time that a thread can yield before needing to be scheduled for execution.

1 3. (Original) A method as recited in claim 1, wherein the device
2 is a battery powered device.

3
4 4. (Previously presented) A method as recited in claim 1,
5 wherein the operating system is a Microsoft WINDOWS CE, Linux,
6 WindRiver, QNX, or PALM operating system.

7
8 5. (Previously presented) A method as recited in claim 1,
9 wherein the predetermined periodic rate is a millisecond.

10
11 6. (Currently amended) A method as recited in claim 1:
12 wherein the providing further comprises setting a system timer to
13 generate a notification at the predetermined periodic rate;

14 wherein the deactivating further comprises resetting the system timer
15 to generate the notification after the dynamic variable amount of time has
16 elapsed since the deactivating; and

17 wherein the method further comprises:

18 receiving the notification after the dynamic variable amount
19 of time has elapsed since the deactivating; and

20 responsive to the receiving:

21 resetting the system timer to generate the notification
22 at the predetermined periodic rate; and

23 activating the the one or more of the hardware modules
24 and the program modules.

1 7. (Canceled).

2
3 8. (Previously presented) A method for providing thread
4 scheduling in a device, the device comprising one or more hardware
5 elements operatively coupled to an operating system comprising a plurality
6 of program modules, the method comprising:

7 scheduling one or more threads at a predetermined periodic rate;

8 determining whether or not there are any threads to execute;

9 responsive to a determination that there are no threads to execute,
10 deactivating one or more of the hardware elements and the program
11 modules for a dynamic variable amount of time, the dynamic variable
12 amount of time being based on a sleep state of a set of threads in a sleep
13 queue and independent of the predetermined periodic rate; and

14 activating the one or more of the hardware elements and the program
15 modules only when the operating system needs to perform an action
16 selected from a group of actions comprising scheduling a thread for
17 execution upon expiration of the dynamic variable amount of time since the
18 deactivating, or upon receipt of an external event that is not a system timer
19 event.

20
21 9. (Original) A method as recited in claim 8, wherein the device
22 comprises a battery powered device.

23
24 10. (Original) A method as recited in claim 8, wherein the
25 operating system comprises a Microsoft WINDOWS CE operating system.

1
2 11. (Previously presented) A method as recited in claim 8,
3 wherein the predetermined periodic rate is a millisecond.
4

5 12. (Original) A method as recited in claim 8:
6 wherein the scheduling further comprises setting a system timer to
7 the predetermined periodic rate, the predetermined periodic rate
8 corresponding to a thread scheduling accuracy; and
9 wherein the deactivating further comprises resetting the system timer
10 to generate a notification after the dynamic variable amount of time has
11 elapsed since the deactivating.
12

13 13. (Original) A method as recited in claim 8:
14 wherein the deactivating further comprises resetting a system timer
15 to generate a notification after the dynamic variable amount of time has
16 elapsed, the dynamic variable amount of time being a maximum amount of
17 time that a thread can yield to other threads before needing to be scheduled
18 for execution; and

19 wherein the activating further comprises resetting the system timer
20 to the predetermined periodic rate to provide substantial thread scheduling
21 accuracy.
22

23 14. (Canceled).
24
25

1 15. (Previously presented) A computer-readable storage medium
2 containing computer-executable instructions for scheduling threads in a
3 device, the device including an operating system comprised of a plurality of
4 program modules that are in turn coupled to one or more hardware
5 elements, the computer-executable instructions comprising instructions for:

6 determining at a periodic rate whether or not there are any threads to
7 execute; and

8 responsive to a determination that there are no threads to execute,
9 deactivating one or more of the program modules and the hardware
10 elements for a dynamic variable amount of time, the dynamic variable
11 amount of time being independent of the periodic rate, the dynamic variable
12 amount of time being based on a sleep state of a set of threads in a sleep
13 queue.
14

15 16. (Original) A computer-readable storage medium as recited in
16 claim 15, wherein the dynamic variable amount of time comprises a
17 maximum amount of time that a thread has specified that it will yield to
18 other threads before it needs to be scheduled for execution.
19

20 17. (Original) A computer-readable storage medium as recited in
21 claim 15, wherein the device comprises a battery powered device.
22

23 18. (Original) A computer-readable storage medium as recited in
24 claim 15, wherein the operating system comprises a Microsoft WINDOWS
25 CE operating system.

1
2 19. (Previously presented) A computer-readable storage medium
3 as recited in claim 15, wherein the computer-executable instructions further
4 comprise instructions for:

5 in the deactivating, configuring a system timer to send a first timer
6 interrupt after the dynamic variable amount of time has elapsed, the
7 dynamic variable amount of time being a maximum amount of time that a
8 first thread can yield to a second thread before the first thread needs to be
9 executed; and

10 responsive to receiving the first timer interrupt:

11 (a) configuring the system timer to send a second timer interrupt at
12 the periodic rate; and

13 (b) activating the one or more of the program modules and the
14 hardware elements to determine if there are any threads to execute.

15
16 20. (Previously presented) A computer-readable storage medium
17 as recited in claim 15, wherein the computer-executable instructions further
18 comprise instructions for:

19 receiving an external interrupt before the dynamic variable amount
20 of time has elapsed since the deactivating, the external interrupt not being a
21 system timer interrupt; and

22 responsive to receiving the external interrupt, processing the external
23 interrupt such that the one or more of the program modules and the
24 hardware elements remain deactivated for the dynamic variable amount of
25 time.

21. (Canceled).

22. (Previously presented) A device comprising:

a processor ;

a plurality of hardware elements coupled to the processor; and a memory coupled to the processor, the memory comprising computer-program instructions executable by the processor, the computer-program instructions comprising a scheduler program module, a hardware abstraction layer (HAL) program module, one or more operating system program modules, and a set of application program modules;

wherein the scheduler comprises computer-executable instructions for:

scheduling threads for execution at a periodic time interval;

determining that there are no threads to execute; and

wherein the HAL, responsive to the determining, comprises computer-executable instructions for deactivating, for a dynamic variable amount of time, one or more of the scheduler, the hardware elements, the one or more operating system program modules, and the application program modules, the dynamic variable amount of time being independent of the periodic time interval and being based on a sleep state of a set of threads in a sleep queue.

1 23. (Original) A device as recited in claim 22, wherein the
2 dynamic variable amount of time is based on a maximum amount of time
3 that a thread can yield before needing to be scheduled.
4

5 24. (Original) A device as recited in claim 22, wherein the
6 periodic time interval is a millisecond.
7

8 25. (Original) A device as recited in claim 22, wherein the device
9 comprises a battery powered device.
10

11 26. (Previously presented) A device as recited in claim 22,
12 wherein the operating system is a Microsoft WINDOWS CE, Linux,
13 WindRiver, QNX, or Palm ® operating system.
14

15 27. (Previously presented) A device as recited in claim 22,
16 wherein the HAL further comprises computer-executable instructions for
17 re-activating the at least one subset of components after the dynamic
18 variable amount of time has elapsed since the one or more of the program
19 modules and the hardware elements were deactivated.
20

21 28. (Original) A device as recited in claim 27, wherein the
22 scheduler is re-activated in a manner that allows the scheduler to schedule
23 threads based on the periodic time interval.
24
25

1 29. (Original) A device as recited in claim 22, wherein after the
2 scheduler is deactivated, the HAL further comprises computer-executable
3 instructions for receiving a notification in response to an external event, the
4 external event not being a system timer event, responsive to receipt of the
5 notification, the HAL processing the notification in a manner that the
6 scheduler remains deactivated for the dynamic variable amount of time.
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25